Problem 1. Suppose that \( a_1, a_2, a_3, \ldots \) and \( b_1, b_2, b_3, \ldots \) are sequences of real numbers. Further, suppose that \( a_i \leq b_i \) for all \( i \in \mathbb{N} \). Prove using induction that for all positive integers \( n \),
\[
\sum_{i=1}^{n} a_i \leq \sum_{i=1}^{n} b_i.
\]

Solution 1. Let \( a_1, a_2, a_3, \ldots \) and \( b_1, b_2, b_3, \ldots \) are sequences of real numbers such that \( a_i \leq b_i \) for all \( i \in \mathbb{N} \). We will prove the claim via induction.

For the base case, it’s given to us that \( a_1 \leq b_1 \), so we need do no more.

Now assume that \( \sum_{i=1}^{n} a_i \leq \sum_{i=1}^{n} b_i \).

For \( n + 1 \), we then have
\[
\sum_{i=1}^{n+1} a_i = \sum_{i=1}^{n} a_i + a_{n+1} \leq \sum_{i=1}^{n} b_i + a_{n+1} \leq \sum_{i=1}^{n} b_i + b_{n+1} = \sum_{i=1}^{n+1} b_i.
\]

Problem 2. Let \( a_n = 4(a_{n-1} - a_{n-2}) \) for \( n \geq 3 \). Suppose that \( a_1 = 2 \) and \( a_2 = 8 \). Prove using induction that for all positive integers \( n \), \( a_n = n2^n \).

Solution 2. For our base cases, we have \( a_1 = 2 = 1*2 = 1*2^1 \) and \( a_2 = 8 = 2*4 = 2*2^2 \).

Now assume that for all \( 1 \leq k \leq n \), we have \( a_k = k2^k \). Consider the \( n + 1 \) case. We have
\[
a_{n+1} = 4(a_n - a_{n-1})
= 4(n2^n - (n - 1)2^{n-1})
= 2^{n+1}(2n - (n - 1))
= (n + 1)2^{n+1}.
\]

Problem 3. Let \( \Sigma \) be the English alphabet. Give a DFA or an NFA which accepts the language
\[
L = (\text{regular})^*.
\]
Solution 3.

Problem 4. Give a DFA or an NFA which accepts the language

\[ L = \{ \omega \in \{0, 1\}^* \mid \text{The last bit and the third to last bit of } \omega \text{ is } 1 \}. \]

Solution 4.

Problem 5. Prove that the finite automaton below accepts the language.

\[ L = \epsilon | (01)(01)(01)^*(0|01))^*. \]
**Solution 5.** It’s easiest to just perform a GNFA reduction to obtain the regular expression for the language.

Problem 6. Let Σ be an alphabet. Given a language \( L \subseteq \Sigma^* \), the odd part of \( L \) is the set

\[ O(L) = \{ \omega \in L \mid |\omega| \text{ is odd} \}. \]

Prove or disprove that if \( L \) is regular, then \( O(L) \) is regular.

**Solution 6.** Suppose \( L \) is regular. Then there is a DFA

\[ D = (Q, \Sigma, \delta, q_0, F) \]

with \( L(D) = L \).

Construct a new DFA

\[ D_o = (Q \times \{E, O\}, \Sigma, \delta_o, (q_0, E), F_o), \]

where \( F_o = \{(q, O) \mid q \in F\} \) and for \((q, P) \in Q \times \{E, O\}\) and \(a \in \Sigma\), we have

\[ \delta_o((q, P), a) = \begin{cases} \delta(q, a), O & P = E \\ \delta(q, a), E & P = O. \end{cases} \]

Observe that \( D_o \) behaves identically to \( D \) except that it’s state space now tracks the parity of its input’s length. So if \( \delta^*(q, \omega) = p \), then we have \( \delta^*_o((q, P), \omega) = (p, P') \), where \( P' \) satisfies the following.

1. If \( P = E \) and \(|\omega| \) is even, then \( P' = E \).
2. If $P = E$ and $|\omega|$ is odd, then $P' = O$.
3. If $P = O$ and $|\omega|$ is even, then $P' = O$.
4. If $P = O$ and $|\omega|$ is odd, then $P' = E$.

We omit a formal inductive proof of this, as it is clear by inspection.

Now observe that $\delta^*(q_0, E, \omega) \in F_0$ iff $|\omega|$ is odd and $\delta^*(q_0, \omega) \in F$. This gives us that $D_o$ only accepts odd length strings in $L$. Moreover, the reverse direction implies that $D_o$ accepts all odd length strings in $L$. Therefore $L(D_o) = O(L)$. Since we have given a DFA for $O(L)$, it follows that $O(L)$ is regular.

Here’s a simple alternate proof. The set of all odd length strings $O(\Sigma^*)$ is obviously regular. Moreover, the class of regular languages is closed under intersection, so if $L$ is regular, then clearly

$$O(L) = O(\Sigma^*) \cap L$$

is also regular.

**Problem 7.** Prove or disprove the converse of the previous problem. That is if $O(L)$ is regular, then $L$ is regular.

**Solution 7.** Consider the language

$$L = \{0^n1^n \mid n \text{ is even}\}.$$ Clearly, $L$ is not regular. Moreover, $O(L) = \emptyset$, which is regular, so the statement is false.

**Problem 8.** We have shown that the class of regular languages is closed under many operations: union, concatenation, Kleene star, intersection, complement, reversal, and perfect shuffle. All of these are finite operations in that they have a finite number of operands. This leaves open the question of closure under infinite operations.

a) Prove that any finite set of strings is regular.

b) A countable union is the union of a countable number of sets, whether finite or countably infinite. For example, the natural numbers is the countable union of the singleton sets $S_n = \{n\}$ for all $n \in \mathbb{N}$.

Prove or disprove that the class of regular languages is closed under countable unions.

**Solution 8.**

a) A finite set of strings $L$ can be written as a finite union of the singleton sets each containing a string of $L$. Since each singleton set is obviously regular, the finite union of these regular sets must also be regular.
b) Pick any irregular language $L$. We can rewrite $L$ as the countable union of the singleton sets each containing a string of $L$. Each of these singletons is regular by the previous problem, but their countable union $L$ is not. Thus the class of regular languages is not closed under countable unions. □

Problem 9. Prove or disprove that the language

$$L = \{0^a1^b \mid 0 < a < b < 4\}$$

is regular.

Solution 9. In the previous problem, we showed that any finite set of strings is regular. $L$ is a finite set of strings, therefore $L$ is regular. □

Problem 10. Prove or disprove that the language

$$L = \{\omega 0^n 1^n \mid \omega \in \{0, 1\}^* \land n > 0\},$$

is regular.

Solution 10. Picking $\omega = \epsilon$ and $\nu_n = \omega 0^n$ will yield an infinite collection of strings that are pairwise distinguishable from each other. Observe that for $m > n$, we have $\nu_m 1^m \in L$ but $\nu_n 1^m \notin L$. As such, it follows that $L$ has an infinite number of equivalence classes with respect to $=_L$. Thus by Myhill-Nerode, $L$ is not regular. □